APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/092,168	03/06/2002	Sridhar Satuloori	5681-08800	9232

7590   02/27/2006

Robert C. Kowert
Conley, Rose, & Tayon, P.C.
P.O. Box 398
Austin, TX 78767

EXAMINER
ZHEN, LI B

ART UNIT	PAPER NUMBER
2194	

DATE MAILED: 02/27/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _19 October 2005_.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1-53_ is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) _1-53_ is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413) Paper No(s)/Mail Date _____.

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____.

WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER

DETAILED ACTION

1.   Claims 1 – 53 are pending in the application.

Response to Arguments

2.   Applicant's arguments with respect to the claims have been considered but are moot in view of the new ground(s) of rejection.

Claim Rejections - 35 USC § 102

3.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4.   Claims 1 – 8, 11 – 21, 24 – 35, 38 – 48 and 51 – 53 are rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent Application Publication No. 2003/0023953 to Lucassen et al. [hereinafter referred to as Lucassen].

5.   As to claim 1, Lucassen teaches a system [MVC-based development system; p. 2, paragraph 0017], comprising:

a processor [p. 7, paragraph 0067];

a computer-accessible medium coupled to the processor, wherein the computer-accessible medium is configured to store program instructions executable by the processor [Tier-2 comprises the business logic that runs on a Web application server, Web server; p. 7, paragraph 0067; examiner notes that the MVC framework includes servers and a processor and memory to store program instructions are inherent to the servers] to implement an application program [a application 50, Fig. 5; p. 11, paragraph 0105] comprising:

one or more application modules [an application data layer 51, a business logic layer 52, an interaction logic layer 53 a customization layer 54, and application process 55; p. 11, paragraph 0105], wherein at least a first one of the application modules comprises a first dynamic component [interaction logic layer 53; p. 11 – 12, paragraph 0107] and a static component [application data layer 51 comprises data content, file services and databases, and comprises all of the backend information; p. 11, paragraph 0105], wherein the first dynamic component and the static component are configured to function according to an initial set of requirements for the application [ability of the system to use the best possible combination of interface modalities based on the user's current preferences, needs and abilities as well as the application requirements and device capabilities; pp. 4 – 5, paragraph 0041]; and

a dynamic component generator configured to receive a new set of requirements [meta-data; p. 12, paragraph 0108] for the application and generate a second dynamic component to replace the first dynamic component [That dynamically generates an interaction logic layer and customization which is then adapted at runtime; p. 3, paragraph 0029], wherein the second dynamic component is configured to function according to the new set of requirements [customization meta-data 54 and generates functional or customized presentations; p. 12, paragraph 0109 and p. 13, paragraph 0123].

6.   As to claim 2, Lucassen teaches the dynamic component generator does not change the static component in response to the new set of requirements [p. 11, paragraph 0105].

7.   As to claim 3, Lucassen teaches the dynamic component generator is configured to generate a second dynamic component to replace the first dynamic component by modifying the first dynamic component in response to the new set of requirements [p. 15, paragraph 0146].

8.   As to claim 4, Lucassen teaches the dynamic component generator is configured to replace the first dynamic component by overwriting the first dynamic component in the computer-accessible medium in response to the new set of requirements [interaction logic (some elements can be added, remove or replaced; p. 12, paragraph 0108].

9.   As to claim 5, Lucassen teaches the new set of requirements is formatted according to an extensible Mark-up Language (XML) schema and stored in the computer-accessible medium [data models 22 (or data type primitives) are XML Schema compliant; p. 6, 0059].

10.   As to claim 6, Lucassen teaches the one or more application modules comprise a second application module comprising a static component and a dynamic component [p. 11, paragraph 0105].

11.   As to claim 7, Lucassen teaches the dynamic component generator is configured to generate a new dynamic component [p. 3, paragraph 0029] for the second application module in response to receiving the new set of requirements [meta-data; p. 12, paragraph 0108].

12.   As to claim 8, Lucassen teaches another dynamic component generator for the dynamic component of the second application module, wherein the other dynamic component generator is configured to generate a new dynamic component for the second application module [p. 3, paragraph 0029] in response to receiving a new set of requirements for the second application module [meta-data; p. 12, paragraph 0108].

13.   As to claim 11, Lucassen teaches the first application module is a model module [Model M; p. 5, paragraph 0044], wherein the static component is a static data model configured to function independent of an application data representation [p. 11, paragraph 0105], and wherein the dynamic component is a dynamic data model

configured to function dependent upon the application data representation [p. 12, paragraph 0109 and p. 13, paragraph 0123] and according to a current set of application requirements [meta-data; p. 12, paragraph 0108] in response to the user input [p. 3, paragraph 0029].

14.   As to claim 12, Lucassen teaches the dynamic data model comprises an Enterprise Java Bean (EJB) entity bean [p. 11, paragraph 0095].

15.   As to claim 13, Lucassen teaches the new set of requirements indicates a change to the application data representation [p. 12, paragraph 0108], and wherein the dynamic component generator is configured to generate a new dynamic data model in response to the change to the application data representation [p. 12, paragraph 0109 and p. 13, paragraph 0123].

16.   As to claim 14, Lucassen teaches a method [p. 2, paragraph 0017], comprising:
        installing one or more application modules [an application data layer 51, a business logic layer 52, an interaction logic layer 53 a customization layer 54, and application process 55; p. 11, paragraph 0105] each comprising a static component [application data layer 51 comprises data content, file services and databases, and comprises all of the backend information; p. 11, paragraph 0105];
        receiving an initial set of requirements for the application modules [meta-data; p. 12, paragraph 0108]; and
        generating one or more initial dynamic components for the one or more application modules [That dynamically generates an interaction logic layer and customization which is then adapted at runtime; p. 3, paragraph 0029], wherein the one or more initial dynamic components are configured to function according to the initial set of requirements [customization meta-data 54 and generates functional or customized presentations; p. 12, paragraph 0109 and p. 13, paragraph 0123].

17.   As to claim 15, Lucassen teaches receiving a new set of requirements for the application modules [meta-data; p. 12, paragraph 0108]; and generating one or more new dynamic components to replace the one or more initial dynamic components [p. 15, paragraph 0146], wherein the one or more new dynamic components are configured to function according to the new set of requirements [interaction logic (some elements can be added, remove or replaced; p. 12, paragraph 0108].

18.   As to claim 16, Lucassen teaches the generating one or more new dynamic components comprises replacing the one or more initial dynamic components by the one or more new dynamic components by modifying the each of the one or more initial dynamic components in response to the new set of requirements [p. 15, paragraph 0146].

19.   As to claim 17, Lucaseen teaches the generating one or more new dynamic components comprises replacing the one or more initial dynamic components by the one or more new dynamic components by overwriting each of the one or more initial dynamic components in a computer-accessible medium in response to the new set of requirements [p. 12, paragraph 0108].

20.   As to claim 18, Lucassen teaches the generating is performed by one or more dynamic component generators, wherein the one or more dynamic component generators are comprised within the same application as the one or more application modules [p. 3, paragraph 0029].

21.   As to claim 19, Lucassen teaches the generating is performed by one or more dynamic component generators comprised within an application server container, wherein the application modules are comprised within the same application server container [interaction components 91, 92 register with the container 92 and the contact between the container 92 and components 90, 91 is programmed in the container 92; p. 15, paragraph 0151].

22.   As to claim 20, Lucassen teaches the generating, the static components comprised by the one or more application modules are not changed in response to the new set of requirements [p. 11, paragraph 0105].

23.   As to claim 21, Lucassen teaches the new set of requirements is formatted according to an extensible Mark-up Language (XML) schema [data models 22 (or data type primitives) are XML Schema compliant; p. 6, 0059].

24.   As to claim 24, Lucassen teaches one of the one or more application modules is a model module [Model M; p. 5, paragraph 0044], wherein the static component is a static data model configured to function independent of an application data representation [p. 11, paragraph 0105], and wherein a dynamic component generated for the one of the one or more application modules is a dynamic data model configured to function dependent upon the application data representation [p. 12, paragraph 0109 and p. 13, paragraph 0123] and according to a current set of requirements [meta-data; p. 12, paragraph 0108] in response to the user input [p. 3, paragraph 0029].

25.   As to claim 25, Lucassen teaches the dynamic data model comprises an Enterprise Java Bean (EJB) entity bean [p. 11, paragraph 0095].

26.   As to claim 26, Lucassen teaches receiving a new set of requirements indicating a change to the application data representation [p. 12, paragraph 0108]; and generating a new dynamic data model in response to the change to the application data representation [p. 12, paragraph 0109 and p. 13, paragraph 0123].

27.   As to claim 27, Lucassen teaches a method [p. 2, paragraph 0017], comprising:
        installing one or more application modules [an application data layer 51, a business logic layer 52, an interaction logic layer 53 a customization layer 54, and application process 55; p. 11, paragraph 0105], wherein at least a first one of the

application modules comprises a first dynamic component [interaction logic layer 53; p. 11 – 12, paragraph 0107] and a static component [application data layer 51 comprises data content, file services and databases, and comprises all of the backend information; p. 11, paragraph 0105], wherein the first dynamic component and the static component are configured to function according to an initial set of requirements for the application [pp. 4 – 5, paragraph 0041];

receiving a new set of requirements for the application modules [meta-data; p. 12, paragraph 0108]; and

generating a new dynamic component to replace the first dynamic component [That dynamically generates an interaction logic layer and customization which is then adapted at runtime; p. 3, paragraph 0029], wherein the new dynamic component is configured to function according to the new set of requirements [customization meta-data 54 and generates functional or customized presentations; p. 12, paragraph 0109 and p. 13, paragraph 0123].

28.   As to claim 28, Lucassen teaches the generating is performed by one or more dynamic component generators, wherein the one or more dynamic component generators are comprised within the same application server as the one or more application modules [p. 3, paragraph 0029].

29.   As to claim 29, Lucassen teaches the generating is performed by one or more dynamic component generator comprised within an application server container, wherein the one or more application modules are comprised within the same application server container [interaction components 91, 92 register with the container 92 and the contact between the container 92 and components 90, 91 is programmed in the container 92; p. 15, paragraph 0151].

30.   As to claim 30, Lucassen teaches in said generating, the static component does not change in response to the new set of requirements [p. 11, paragraph 0105].

31.   As to claim 31, Lucassen teaches in said generating, the second dynamic component replaces the first dynamic component by modifying the first dynamic component in response to the new set of requirements [p. 15, paragraph 0146].

32.   As to claim 32, Lucassen teaches in said generating, the second dynamic component replaces the first dynamic component by overwriting the first dynamic component in a computer-accessible medium in response to the new set of requirements [p. 12, paragraph 0108].

33.   As to claim 33, Lucassen teaches wherein the new set of requirements is formatted according to an extensible Mark-up Language (XML) schema and stored in the computer-accessible medium [data models 22 (or data type primitives) are XML Schema compliant; p. 6, 0059].

34.   As to claim 34, Lucassen teaches wherein the one or more application modules comprise a second application module comprising a static component and a dynamic component [p. 11, paragraph 0105].

35.   As to claim 35, Lucassen teaches generating a new dynamic component [p. 3, paragraph 0029] for the second application module in response to receiving the new set of requirements [meta-data; p. 12, paragraph 0108].

36.   As to claims 38 – 40, these are similar in scope to claims 11 – 13; therefore, they are rejected for the same reasons as claims 11 – 13 above.

37.   As to claims 41 – 48 and 51 – 53, these are product claims that correspond to system claims 1 – 8 and 11 – 13; note the rejections to claims 1 – 8 and 11 – 13 above, which also meet these product claims.

Claim Rejections - 35 USC § 103

38.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set
> forth in section 102 of this title, if the differences between the subject matter sought to be patented and
> the prior art are such that the subject matter as a whole would have been obvious at the time the
> invention was made to a person having ordinary skill in the art to which said subject matter pertains.
> Patentability shall not be negatived by the manner in which the invention was made.

39.   Claims 9, 10, 22, 23, 36, 37, 49 and 50 are rejected under 35 U.S.C. 103(a) as

being unpatentable over Lucassen in view of U.S. Patent Application Publication No.

2002/0109734 to Umezu et al. [hereinafter referred to as Umezu].

40.   As to claim 9, Lucassen teaches the first application module is a controller

module [Controllers C1, C2 and C3; p. 2, paragraph 0014], wherein the dynamic

component is an application logic component coupled [interaction logic layer 53; p. 11 –

12, paragraph 0107], wherein the application logic component is configured to function

according to a current set of application requirements in response to the user input

[customization meta-data 54 and generates functional or customized presentations; p.

12, paragraph 0109 and p. 13, paragraph 0123]. Lucassen does not disclose the static

component as a router component configured to receive user input.

        However, Umezu teaches model view controller includes allocating functions of

GUI to an object (model) which stores a service, which is a core function of an

application, and data, which is base of display, an object (view) for performing display,

and an object (controller) receiving input from an input device, and operates GUI

through cooperation of the view and the controller [p. 4, paragraph 0068], and a static

component is a router component configured to receive user input [p. 14, paragraphs

0333 – 0336]

41.   It would have been obvious to a person of ordinary skill in the art at the time of

the invention to apply the teaching of a router component configured to receive user

input as taught by Umezu to the invention of Lucassen because this provides a GUI

processing system which enables an operation of drag-and-drop providing visual

feedback, allows undo of an operation of application with GUI, and allows redo of an undone operation of application with GUI [p. 2, paragraph 0021 of Umezu].

42.   As to claim 10, Lucassen teaches the application logic component comprises an Enterprise Java Bean (EJB) session bean [p. 11, paragraph 0095].

43.   As to claim 22, Lucassen as modified teaches one of the one or more application modules is a controller module [Controllers C1, C2 and C3; p. 2, paragraph 0014 of Lucassen], wherein the static component is a router component configured to receive user input [Controllers C1, C2 and C3; p. 2, paragraph 0014 of Umezu], and wherein a dynamic component generated for the one of the one or more application modules is an application logic component coupled to the router component [p. 4, paragraph 0068 of Umezu], wherein the application logic component is configured to function according to a current set of requirements in response to the user input [customization meta-data 54 and generates functional or customized presentations; p. 12, paragraph 0109 and p. 13, paragraph 0123 of Lucassen].

44.   As to claim 23, Lucassen teaches the application logic component comprises an Enterprise Java Bean (EJB) session bean [p. 11, paragraph 0095].

45.   As to claims 36 and 37, these are similar in scope to claims 22 and 23; therefore, they are rejected for the same reasons as claims 22 and 23 above.

46.   As to claims 49 – 50, these are product claims that correspond to system claims 9 and 10; note the rejection to claims 9 and 10 above, which also meet these product claims.

CONTACT INFORMATION

47.   Any inquiry concerning this communication or earlier communications from the examiner should be directed to Li B. Zhen whose telephone number is (571) 272-3768. The examiner can normally be reached on Mon - Fri, 8:30am - 5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, William Thomson can be reached on 571-272-3718. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Li B. Zhen
Examiner
Art Unit 2194

lbz

WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER